



**Helion**

Zbuduj własne urządzenia pomiarowe z platformą Arduino!

# Monitorowanie otoczenia z Arduino

Emily Gertz, Patrick Di Justo



**MAKERMEDIA™**

Tytuł oryginału: Environmental Monitoring with Arduino:  
Building Simple Devices to Collect Data About the World Around Us

Tłumaczenie: Mikołaj Szczepaniak

ISBN: 978-83-246-8168-6

© 2013 Helion S.A.

Authorized Polish translation of the English edition of Environmental Monitoring With Arduino, ISBN 9781449310561 © 2012 Emily Gertz and Patrick Di Justo published by Maker Media Inc.

This translation is published and sold by permission of O'Reilly Media, Inc., which owns or controls all rights to sell the same.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Wydawnictwo HELION dołożyło wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie bierze jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Wydawnictwo HELION nie ponosi również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Wydawnictwo HELION  
ul. Kościuszki 1c, 44-100 GLIWICE  
tel. 32 231 22 19, 32 230 98 63  
e-mail: [helion@helion.pl](mailto:helion@helion.pl)  
WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Pliki z przykładami omawianymi w książce można znaleźć pod adresem:  
<ftp://ftp.helion.pl/przyklady/monito.zip>

Drogi Czytelniku!  
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres  
<http://helion.pl/user/opinie/monito>  
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

# Spis treści

|   |           |
|---|-----------|
| <b>Przedmowa</b> .....  | <b>9</b>  |
| <b>1/Najkrótszy na świecie podręcznik do elektroniki</b> .....                            | <b>15</b> |
| Czym jest Arduino? .....  | 15        |
| Obwody i komponenty elektroniczne .....   | 17        |
| Programowanie platformy Arduino .....   | 20        |
| Pierwszy szkic: migająca dioda LED .....  | 20        |
| Potrzebne części .....  | 21        |
| Instalacja środowiska IDE .....   | 21        |
| Budowa obwodu .....   | 21        |
| Pisanie kodu .....  | 21        |
| Dalsze eksperymenty .....   | 23        |
| <b>2/Projekt: monitor hałasu i wyświetlanie informacji na taśmie LED</b> .....            | <b>25</b> |
| Mierzenie poziomu hałasu — mikrofon .....   | 25        |
| Taśma diod LED .....  | 27        |
| Budowa gadżetu .....  | 28        |
| Potrzebne części .....  | 28        |
| Obwód na płytce uniwersalnej .....  | 28        |
| Pisanie kodu .....  | 30        |
| Dalsze eksperymenty .....   | 32        |
| <b>3/Nowy komponent: wyświetlacz 4-znakowy</b> .....                                      | <b>33</b> |
| Projekt testowy .....   | 34        |
| Potrzebne części .....  | 34        |
| Obwód na płytce uniwersalnej .....  | 34        |
| Pisanie kodu .....  | 35        |
| Dalsze eksperymenty .....   | 38        |
| <b>4/Interferencja elektromagnetyczna (i generowanie naprawdę kiepskiej muzyki)</b> ..... | <b>39</b> |
| Wykrywanie źródeł EMI w środowisku .....  | 40        |
| Budowa gadżetu .....  | 40        |
| Potrzebne części .....  | 41        |
| Głośnik 8-omowy .....   | 41        |
| Konstrukcja monitora promieniowania EMI .....   | 43        |
| Pisanie kodu .....  | 44        |

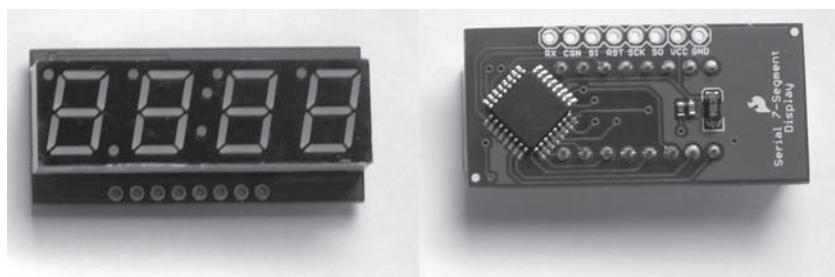
|  |           |
|--|-----------|
| Uruchamianie szkicu .....  | 45        |
| Zasilanie gadżetu w trybie mobilnym .....  | 46        |
| Co można mierzyć za pomocą tego gadżetu? .....   | 46        |
| Dalsze eksperymenty .....  | 47        |
| <b>5/Projekt: mierzenie przewodności wody i numeryczne dane wyjściowe .....</b>                            | <b>49</b> |
| Czym jest przewodność i dlaczego jest taka ważna? .....  | 49        |
| Budowa gadżetu .....   | 49        |
| Potrzebne części .....   | 50        |
| Budowa sondy .....   | 51        |
| Obwód na płytce uniwersalnej .....   | 52        |
| Pisanie kodu .....   | 53        |
| Jak wykonywać pomiary? .....   | 54        |
| Dalsze eksperymenty .....  | 54        |
| <b>6/Nowy komponent: moduł sieciowy .....</b>  | <b>57</b> |
| Stosowanie modułu sieciowego .....   | 59        |
| Port sieciowy .....  | 59        |
| Adres MAC .....  | 59        |
| Adres IP .....   | 59        |
| Gniazdo kart SD .....  | 61        |
| Testowanie modułu sieciowego .....   | 62        |
| Potrzebne części .....   | 63        |
| Budowa układu .....  | 63        |
| Testowanie gniazda kart SD .....   | 63        |
| Potrzebne części .....   | 63        |
| Budowa układu .....  | 64        |
| Dalsze eksperymenty .....  | 64        |
| <b>7/Projekt: wyświetlanie wilgotności, temperatury i punktu rosy<br/>na wyświetlaczu 4-znakowym .....</b> | <b>65</b> |
| Nie trzeba być meteorologiem, aby badać stan pogody .....  | 65        |
| Gromadzenie wiarygodnych pomiarów .....  | 66        |
| Pierwszy czujnik elektroniczny: DHT-22 .....   | 68        |
| Stosowanie bibliotek kodu .....  | 69        |
| Budowa gadżetu .....   | 70        |
| Potrzebne części .....   | 70        |
| Obwód na płytce uniwersalnej .....   | 70        |
| Pisanie kodu .....   | 71        |
| Dalsze eksperymenty .....  | 75        |

|   |           |
|---|-----------|
| <b>8/Udostępnianie w czasie rzeczywistym geotagowanych danych<br/>za pośrednictwem serwisu Xively .....</b> | <b>77</b> |
| Projekt testowy: nawiązywanie połączenia z serwisem Xively<br>i wysyłanie do niego danych .....             | 78        |
| Potrzebne części .....  | 78        |
| Rejestracja konta w serwisie Xively .....   | 78        |
| Pisanie kodu .....  | 79        |
| Dalsze eksperymenty .....   | 80        |
| <b>9/Mierzenie promieniowania i udostępnianie danych w internecie .....</b>                                 | <b>81</b> |
| Czym jest licznik Geigera? .....  | 82        |
| Budowa gadżetu .....  | 84        |
| Potrzebne części .....  | 84        |
| Obwód na płytce uniwersalnej .....  | 85        |
| Pisanie kodu .....  | 86        |
| Co można mierzyć za pomocą tego gadżetu? .....  | 90        |
| Analiza w razie awarii .....  | 91        |
| Dalsze eksperymenty .....   | 92        |
| <b>10/Obudowa dla gadżetu .....</b>   | <b>93</b> |
| <b>Skorowidz .....</b>  | <b>94</b> |



# 3/Nowy komponent: wyświetlacz 4-znakowy

W następnym projekcie dane będą prezentowane za pomocą 4-znakowego wyświetlacza LED sterowanego szeregowo (zob. rysunek 3.1). To wyjątkowo wszechstronne narzędzie składa się z czterech 7-segmentowych wyświetlaczy LED umożliwiających wyświetlanie — jak nietrudno zgadnąć — czterech znaków jednocześnie (a także dwukropka i znaków dziesiętnych).



Rysunek 3.1. 4-znakowy wyświetlacz LED (z lewej strony pokazano przód wyświetlacza, z prawej — tył)

Wyświetlacz 4-znakowy może prezentować wszystkie cyfry arabskie (od 0 do 9), a także 20 spośród 26 liter alfabetu łacińskiego (niektóre zarówno w formie wielkiej, jak i małej litery). Przykłady liter na wyświetlaczu tego typu pokazano na rysunku 3.2. (Istnieją pewne konwencje dotyczące wyświetlania takich liter jak *k*, *m*, *v*, *w* czy *x*, jednak nawet stosowanie tych standardów nie rozwiązuje problemu błędnego rozpoznawania wymienionych liter przez większość ludzi — niektóre z nich sprawiają wrażenie losowo podświetlanych segmentów i są interpretowane raczej jako efekt błędów).



Rysunek 3.2. Sposób wyświetlania liter alfabetu łacińskiego na wyświetlaczu 4-znakowym

O przydatności wyświetlaczy tego typu w dużej mierze decyduje możliwość sterowania szeregowo. W standardowym wyświetlacz 7-segmentowym każdy segment wymaga sterowania za pośrednictwem odrębnej, dedykowanej linii zajmującej jeden z pinów na płytce Arduino. Stosowanie czterech wyświetlaczy tego typu (plus punkty dziesiętne i dwukropek) wymagałoby aż 34 dedykowanych linii danych, czyli więcej, niż jest dostępne na płytce Arduino. (Na szczęście istnieją rozwiązania umożliwiające obejście tego problemu, ale nawet one wymagają stosunkowo dużej liczby linii danych).

Właśnie dlatego konstruktorzy wyświetlacza 4-znakowego z firmy SparkFun umieścili za właściwym wyświetlaczem mikrokontroler. Mikrokontroler odpowiada za odbieranie danych szeregowych wysyłanych za pośrednictwem jednego pinu na płytce Arduino i interpretację tych danych w celu prawidłowego wyświetlenia wszystkich czterech znaków.

Każde ułatwienie ma jednak swoją cenę. W tym przypadku ułatwieniem jest możliwość sterowania 4-znakowym wyświetlaczem LED za pomocą dwóch dedykowanych linii danych, a ceną jest konieczność takiego formatowania danych, aby były każdorazowo wysyłane w grupach po 4 znaki. Zawsze. Nie można liczyć na żadne wyjątki. Chcesz wyświetlić tylko 3 znaki? To niczego nie zmienia. Tak czy inaczej musisz dodać spację lub inny znak, aby wyświetlacz otrzymał pełen pakiet 4 znaków. W razie konieczności wyświetlenia co najmniej 5 znaków należy tak sformatować dane, aby trafiły do wyświetlaczy w 4-znakowych pakietach i aby znaki, które nie mieszczą się na pierwszym wyświetlaczu, były prawidłowo przekazywane do drugiego i kolejnych komponentów.

## Projekt testowy

Aby lepiej zrozumieć działanie wyświetlacza 4-znakowego, warto zbudować prosty obwód i napisać przykładowy kod.

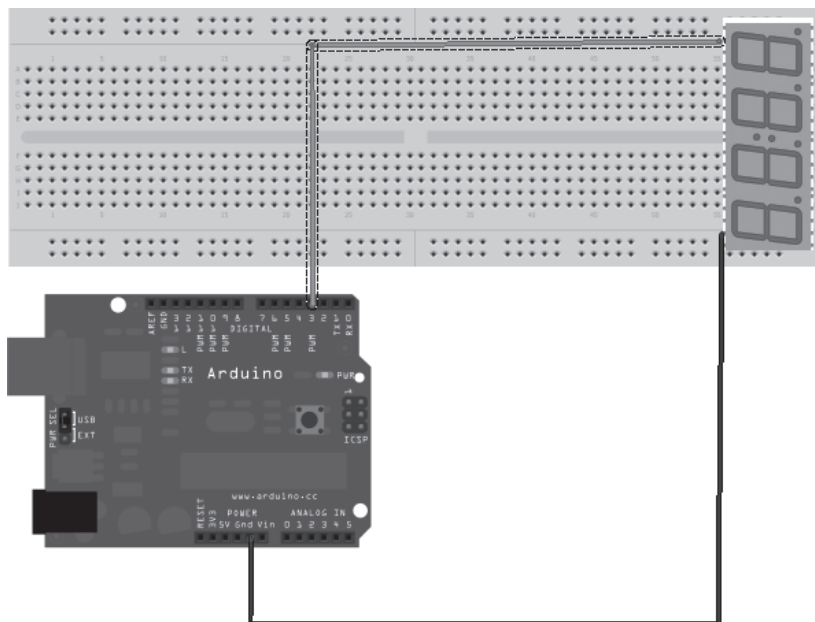
### Potrzebne części

1. Arduino
2. Płytki uniwersalna
3. Wyświetlacz 4-znakowy (produkt COM-09765 w sklepie SparkFun)
4. Przewody połączeniowe różnych kolorów

### Obwód na płytce uniwersalnej

Budowa tego obwodu jest bardzo prosta, co widać na rysunku 3.3, na którym pokazano jego strukturę na płytce uniwersalnej.





Rysunek 3.3. Kompletny obwód do testów wyświetlacza 4-znakowego

**Krok 1.** Podłączenie przewodu pomiędzy pinem GND na płytce Arduino a pinem GND wyświetlacza 4-znakowego.

**Krok 2.** Instalacja przewodu łączącego pin uziemienia wyświetlacza z odpowiednim otworem na płytce uniwersalnej. (Wybór wiersza płytki uniwersalnej zależy od konstruktora. Nie należy jednak stosować wiersza rozciągającego się na całą szerokość płytki).

**Krok 3.** Połączenie pinu cyfrowego nr 3 na płytce Arduino z pinem RX wyświetlacza 4-znakowego. Także w tym przypadku przewód należy podłączyć do pinu RX wyświetlacza za pośrednictwem otworów płytki uniwersalnej.

## Pisanie kodu

Poniższy przykładowy szkic zawiera (w komentarzach) wiele wskazówek dotyczących sterowania wyświetlaczem 4-znakowym. Kod dla tego projektu można znaleźć w repozytorium GitHub pod adresem <https://github.com/ejgertz/EMWA/blob/master/chapter-3/4Char>.

Szkic należy wysłać na platformę Arduino.

```
/*
  Test wyświetlacza 4-znakowego
  Ten szkic ilustruje techniki przewijania tekstu na wyświetlaczu 4-znakowym.
  Zgodnie z tradycją pierwszy program powinien wyświetlać komunikat Witaj świecie,
  jednak wyświetlacz 4-znakowy nie pozwala wyświetlać np. litery w. W tej sytuacji musimy improwizować...
  Kod dla tego przykładu przygotowano na bazie rozwiązań dostępnych publicznie.
  */
```

```

#include <SoftwareSerial.h>

#define SerialIn 2
#define SerialOut 3
#define Xdelay 600

String txtmessage = "HeJ JeSt SuPer";

byte thou=0;
byte hund=0;
byte tens=0;
byte ones=0;

SoftwareSerial mySerialPort(SerialIn, SerialOut);

void setup()
{
  pinMode(SerialOut, OUTPUT);
  pinMode(SerialIn, INPUT);

  // rozpoczyna komunikację z wyświetlaczem 4-znakowym
  mySerialPort.begin(9600);

  // znak 'v' resetuje wyświetlacz
  mySerialPort.print("v");
}

void loop()
{
  // włącza wszystkie segmenty w celach testowych
  mySerialPort.print("----");
  delay(Xdelay);
  mySerialPort.print("8888");
  delay(Xdelay);
  mySerialPort.print("----");
  delay(Xdelay);
  mySerialPort.print("xxxx");
  delay(Xdelay);

  // przewija cyfry od 1 do 0 w prosty, ale mało ciekawy sposób
  mySerialPort.print("xxxx");
  delay(Xdelay);
  mySerialPort.print("xxx1");
  delay(Xdelay);
  mySerialPort.print("xx12");
  delay(Xdelay);
  mySerialPort.print("x123");
  delay(Xdelay);
  mySerialPort.print("1234");
  delay(Xdelay);
  mySerialPort.print("2345");
  delay(Xdelay);
  mySerialPort.print("3456");
  delay(Xdelay);
  mySerialPort.print("4567");
  delay(Xdelay);
  mySerialPort.print("5678");
}

```

```

delay(Xdelay);
mySerialPort.print("6789");
delay(Xdelay);
mySerialPort.print("7890");
delay(Xdelay);
mySerialPort.print("8900");
delay(Xdelay);
mySerialPort.print("9000");
delay(Xdelay);
mySerialPort.print("0000");
delay(Xdelay);

// liczy od -1009 do 2000
for(int i = -1009; i<2000; i++)
{
    if((i<-999) || (i>9999))
    {
        mySerialPort.print("ERRx");
        return;
    }
    char fourChars[5];
    sprintf(fourChars, "%04d", i);

    mySerialPort.print("v");
    mySerialPort.print(fourChars);

    // jeśli liczby zmieniają się zbyt szybko, wystarczy zwiększyć opóźnienie
    // delay(Xdelay);
}

delay(Xdelay);
mySerialPort.print("xxxx");
delay(Xdelay);

// przewija wartość zmiennej txtmessage w dość skomplikowany sposób
// dodaje najpierw niezbędny bufor
txtmessage = "xxxx"+txtmessage+"xxxxx";
// następnie konwertuje łańcuch (obiekt typu String) na tablicę znaków (typu char)
// dzięki temu będzie możliwe wyświetlanie danych za pomocą biblioteki SoftwareSerial
char temps[txtmessage.length()];
txtmessage.toCharArray(temps,txtmessage.length());

// wyświetlany tekst należy jeszcze przewinąć
for(int i = 0; i <= txtmessage.length()-5; i++)
{
    mySerialPort.print(temps[i]);
    mySerialPort.print(temps[i+1]);
    mySerialPort.print(temps[i+2]);
    mySerialPort.print(temps[i+3]);
    delay(Xdelay);
}

delay(Xdelay);
delay(Xdelay);
delay(Xdelay);
}

```

Działa? Dobra robota! Właśnie udało nam się zaprogramować przewijanie tekstu na wyświetlaczu 4-znakowym

## Dalsze eksperymenty

1. Instrukcja użytkownika wyświetlacza 4-znakowego LED przygotowana przez firmę SparkFun zawiera wskazówki, jak sterować poszczególnymi segmentami wyświetlacza, a także przecinkiem, apostrofem i punktami dziesiętnymi. Czy potrafisz wykorzystać tę wiedzę do stworzenia licznika odliczającego od 59:59 do 00:00?
2. Przypuśćmy, że z jakiegoś powodu musimy wyświetlać litery *k*, *m*, *v*, *w* lub *x*. Jeśli na przykład system musi wyświetlić takie słowa jak *kamyk*, *Xawier* czy *Volvo* (choć nie do końca wiemy, dlaczego miałbyś to robić), jak zrealizowałbyś to zadanie? Które segmenty należałoby podświetlić?

# Skorowidz

## A

adres  
  IP routera, 59–61  
  MAC modułu sieciowego, 59  
API, application programming interfaces, 77  
Arduino, 15  
  Uno, 16

## B

biblioteka kodu, 69

## C

CPM, counts per minute, 90  
CRS, Cotton Region Shelter, 66  
czujnik  
  elektroniczny DHT-22, 68  
  hałasu, 29

## D

dane  
  geotagowane, 77  
  o pogodzie, 77  
dioda LED, 17  
dodawanie komentarzy, 24  
działanie  
  licznika Geigera, 83  
  wyświetlacza 4-znakowego, 34

## E

EMI, electromagnetic interference, 39

## F

funkcja  
  loop(), 20  
  setup(), 20

## G

głośnik, 41  
gniazdo kart SD, 61, 63

## I

IDE, integrated development environment, 21  
instalacja środowiska IDE, 21  
interfejs API, 77  
interferencja elektromagnetyczna, EMI, 39  
IP, Internet Protocol, 59

## J

jednostka CPM, 90

## K

kalibracja sondy, 55  
karta  
  microSD, 62  
  SD, 61  
klatka meteorologiczna, 66  
klimat, 67  
kod  
  monitora hałasu, 30  
  monitora przewodności wody, 53  
  monitora temperatury, wilgotności i punktu rosy, 71  
  odpowiedzialny za współpracę z serwisem Xively, 86  
  wykrywacza promieniowania EMI, 44  
  wyświetlacza 4-znakowego, 35  
komentarze, 22, 24  
konto w serwisie Xively, 78

## L

liczba impulsów na minutę, 90  
licznik  
    Geigera, 81–84, 91  
    Sensitive Geiger Counter, 82  
lutowanie, 18

## M

MAC, Media Access Control, 59  
mierzenie  
    poziomu hałasu, 25  
    promieniowania, 81, 84, 90  
    przewodności wody, 49  
    temperatury, 66  
mikrofon  
    Emartee Mini Sound Sensor, 26  
    Mini Sound Sensor, 26  
moduł sieciowy, Ethernet shield, 57, 62  
moduły, shields, 57  
monitor hałasu, 25

## N

numeryczne dane wyjściowe, 49

## O

obudowa, 93  
obwód  
    do mierzenia temperatury, wilgotności i  
    punktu rosy, 70  
    elektroniczny, 17  
    licznika Geigera, 85  
    monitora hałasu, 28  
    sondy, 52  
    wyświetlacza 4-znakowego, 35  
ograniczanie zużycia energii, 40  
opcje sieciowe, 61

## P

plik pitches.h, 42  
płytki  
    Arduino, 23  
    Arduino Uno, 16  
    uniwersalna, 18

połączenia sieciowe, 60  
połączenie z serwisem Xively, 78  
port sieciowy, 59  
programowanie platformy, 20  
promieniowanie, radiation, 41  
    EMI, 41  
    jądrowe, 81  
przewodność, 49  
przewód, 19  
punkt rosy, 66

## R

repozytorium GitHub, 21  
rezystor, 17  
router, 60

## S

serwis  
    Instructables, 93  
    MAKE Magazine, 93  
    Maker Shed, 19  
    SciStarter, 68  
    Smarter Living, 40  
    Xively, 77–79, 92  
        nawiązywanie połączenia, 78  
        rejestracja konta, 78  
        testowanie połączenia, 79  
sklepy internetowe, 19, 93  
sonda, 49, 51  
szkic  
    migająca dioda LED, 20  
    ToneMelody, 42  
szkice, sketches, 20

## Ś

środowisko  
    Arduino IDE, 22  
    IDE, 21

## T

taśma diod LED, 27  
testowanie  
    gniazda kart SD, 63  
    modułu sieciowego, 62  
    połączenia z serwisem, 79

## W

wykrywacz promieniowania EMI, 43  
wykrywanie źródeł EMI, 40  
wyświetlacz 4-znakowy, 33, 65  
wyświetlanie  
  na taśmie LED, 27  
  na wyświetlaczu 4-znakowym, 33  
  punktu rosy, 65  
  temperatury, 65  
  wilgotności, 65

## Z

zanieczyszczenie hałasem, 25  
zastosowania  
  modułu sieciowego, 59  
  wykrywacza promieniowania EMI, 46  
zintegrowane środowisko programowania,  
  IDE, 21

## Ź

źródła EMI, 40





# PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW  
w działający bankomat!

**Dowiedz się więcej i dołącz już dzisiaj!**

<http://program-partnerski.helion.pl>

GRUPA WYDAWNICZA

 **Helion SA**

## Arduino — Twoja przepustka do świata elektroniki!

Arduino to niesamowita platforma, która otworzyła przed światem elektroniki mnóstwo nowych możliwości. Dzięki prostocie obsługi zdobyła rzeszę zwolenników, a to przełożyło się na ilość dostępnych w sieci materiałów i publikacji. Dla platformy Arduino opracowano liczne akcesoria, które pozwalają użytkownikowi rozbudować każdy układ. Dodatkowo istnieje możliwość połączenia jej z kolejną popularną platformą — Raspberry Pi. Z tym duetem osiągniesz wszystko!

Ten fantastyczny podręcznik pokaże Ci, jak wykorzystać Arduino do zbierania informacji o otaczającym świecie. Pomiar poziomu hałasu, temperatury i wilgotności to tylko część projektów, które możesz zrealizować dzięki tej platformie. W trakcie lektury dowiesz się, jak wykorzystać moduł sieciowy oraz zaprezentować zebrane dane na wyświetlaczu. Chciałbyś mieć swój własny licznik Geigera? To cudo jest w Twoim zasięgu! Podczas realizacji przedstawionych tu projektów poznasz platformę Arduino i zaczniesz tworzyć własne układy. Sięgnij po tę książkę i zanurz się w świat układów elektronicznych.

### Zbuduj swój własny:

- termometr
- miernik hałasu
- licznik Geigera
- wymarzony układ

### z platformą Arduino!

sięgnij po **WIĘCEJ!**



KOD KORZYSCI

cena: 27,00 zł

**helion.pl**  
księgarnia  
internetowa

Nr katalogowy: 16325



Księgarnia internetowa  
<http://helion.pl>



Zamówienia telefoniczne:  
**0 801 339900**



**0 601 339900**

**MAKERMEDIA**



**Helion**

Sprawdź najnowsze promocje:  
• <http://helion.pl/promocje>  
Książki najchętniej czytane:  
• <http://helion.pl/bestsellery>  
Zamów informacje o nowościach:  
• <http://helion.pl/nowosci>

Helion SA  
ul. Kościuszki 1c, 44-100 Gliwice  
tel.: 32 230 98 63  
e-mail: [helion@helion.pl](mailto:helion@helion.pl)  
<http://helion.pl>

ISBN 978-83-246-8168-6



9 788324 681686

Informatyka w najlepszym wydaniu